UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/776,267	02/02/2001	James J. Fallon	8011-15	9730

22150	7590	06/06/2005	

F. CHAU & ASSOCIATES, LLC
130 WOODBURY ROAD
WOODBURY, NY 11797

EXAMINER
SURYAWANSHI, SURESH

ART UNIT	PAPER NUMBER
2115 .	

DATE MAILED: 06/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	09/776,267	FALLON ET AL.
	Examiner	Art Unit
	Suresh K. Suryawanshi	2115

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _02 May 2005_.
2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1,2,4-7,9,10,12,13,15 and 17_ is/are pending in the application.
   4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) _1,2,4-7,9,10,12,13,15 and 17_ is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☒ The drawing(s) filed on _02 February 2001_ is/are: a)☐ accepted or b)☒ objected to by the Examiner.
   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a)☐ All   b)☐ Some * c)☐ None of:
     1.☐ Certified copies of the priority documents have been received.
     2.☐ Certified copies of the priority documents have been received in Application No. _____.
     3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
        application from the International Bureau (PCT Rule 17.2(a)).
   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)
1)☒ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.
4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____ .
5)☐ Notice of Informal Patent Application (PTO-152)
6)☐ Other: _____.

U.S. Patent and Trademark Office
PTOL-326 (Rev. 1-04)   Office Action Summary   Part of Paper No./Mail Date 3

DETAILED ACTION

1.   Claims 1-2, 4-7, 9-10, 12-13, 15 and 17 are presented for examination.

Drawings

2.   This application, filed under former 37 CFR 1.60, lacks formal drawings. The informal

drawings filed in this application are acceptable for examination purposes. When the application

is allowed, applicant will be required to submit new formal drawings. In unusual circumstances,

the formal drawings from the abandoned parent application may be transferred by the grant of a

petition under 37 CFR 1.182.

Claim Rejections - 35 USC § 103

1.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

2.   Claims 1-2, 4-7, 9-10, 12-13, 15 and 17 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Kroeker et al (US Patent no 6,073,232[1]) in view of Esfahani et al (US Patent

no 6,434,695 B1[1]).

---

[1] Prior art cited by the examiner in the prior office action.

3.   As per claim 1, Kroeker et al teach

maintaining a list of boot data used for booting a computer system [col. 2, lines 30-47;

col. 5, lines 1-7; a prefetch table containing a listing of the disk locations and length of data

records that were requested by the host computer in the immediately previous power-on/reset];

initializing a central processing unit of the computer system [col. 2, lines 30-35; inherent

to the system during power-up process];

preloading the boot data into a cache memory prior to completion of initialization of the

central processing unit of the computer system [col. 1, lines 58-64; col. 2, lines 36-41; col. 3,

lines 30-39; col. 5, lines 17-21; data is preloaded into the RAM cache according to the prefetch

table prior to completion of initialization of the central processor unit as shown in Fig. 3 that the

method enters an idle state to await a command from the host computer since the CPU of the

host computer is not ready]; and

servicing requests for boot data from the computer system using the preloaded boot data

after completion of initialization of the central processing unit of the computer system [col. 2,

lines 41-47; col. 3, lines 30-39; data is communicated from the cache to the host computer as

soon as the host computer requests the data upon completion of initialization of the CPU].

Kroeker et al do not disclose about accessing compressed boot data and decompressing the compressed boot data. However, Esfahani et al clearly disclose about loading a compressed boot data into a RAM cache and then the boot data is decompressed and executed [col. 2, lines 5-13, 63, 67; col. 10, line 65 – col. 11, line 4]. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the cited references as both are directed to minimize a computer's initial program load time or shortening the load time of the computer programs from a hard disk drive to a host computer. Moreover, the shortening load time method of Kroeker et al by loading the program codes into the RAM cache according to the prefetch table will definitely be benefited with the method of reading compressed data into the RAM cache and then decompressing and executing as needed. This way, one may not only have needed data into a fast access memory but also a large amount of data to avoid frequent accessing the storage device(s).

4.   As per claim 2, Kroeker et al teach that the boot data comprises program code associated with one of an operating system of the computer system, an application program, and a combination thereof [col. 5, lines 41-51; requesting data records are part of a computer program such as DOS or Windows].

5.   As per claim 4, Kroeker et al teach that the method steps are performed by a data storage controller connected to the boot device [fig. 1; controller].

6.   As per claim 5, Kroeker et al teach the step of updating the list of boot data during the

boot process [col. 8, lines 63-65; the prefetch table is updated].

7.   As per claim 6, Kroeker et al teach the step of updating comprises adding to the list any

boot data requested by the computer system not previously stored in the list [col. 8, lines 63-68;

the prefetch table is updated].

8.   As per claim 7, Kroeker et al teach that the step of updating comprises removing from the

list any boot data previously stored in the list and not requested by the computer system [col. 8;

lines 63-65; updating the prefetch table].

9.   As per claims 9 and 12, Kroeker et al teach that the method steps are program

instructions that are tangibly embodied on a program storage device and readable by a machine

to execute the method steps [col. 9, lines 27-30; computer program].

10.   As per claim 10, Kroeker et al teach

        maintaining a list of application data associated with an application program [col. 11;

lines 30-34; a prefetch table containing disk storage location and length of the data records

requested by the application program];

preloading the application data upon launching the application program [col. 11, lines 46-50; preloading the data cache prior to receiving a read command from the application]; and

servicing requests for application data from a computer system using the preloaded application data [col. 11, lines 51-57; communicating the prestored data records of the application from the data cache to the host computer].

Kroeker et al do not disclose about accessing compressed data and decompressing the compressed data. However, Esfahani et al clearly disclose about loading a compressed data into a RAM cache and then the compressed data is decompressed and executed [col. 2, lines 5-13, 63, 67; col. 10, line 65 – col. 11, line 4]. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the cited references as both are directed to minimize a computer's initial program load time or shortening the load time of the computer programs from a hard disk drive to a host computer. Moreover, the shortening load time method of Kroeker et al by loading the program codes into the RAM cache according to the prefetch table will definitely be benefited with the method of reading compressed data into the RAM cache and then decompressing and executing as needed. This way, one may not only have needed data into a fast access memory but also a large amount of data to avoid frequent accessing the storage device(s).

11.   As per claim 13, Kroeker et al teach

a digital signal processor (DSP) [fig. 1; host computer];

a programmable volatile logic device [fig. 1, RAM cache], wherein the programmable

volatile logic device is programmed by the DSP or controller prior to completion of initialization

of a central processing unit of the host system [col. 1, lines 58-64; col. 2, lines 36-41; col. 3, lines

30-39; col. 5, lines 17-21; data is preloaded into the RAM cache according to the prefetch table

prior to completion of initialization of the central processor unit as shown in Fig. 3 that the

method enters an idle state to await a command from the host computer since the CPU of the

host computer is not ready] to (i) instantiate a first interface for operatively interfacing the boot

device controller to a boot device [fig. 1; controller] and to (ii) instantiate a second interface for

operatively interfacing the boot device controller to the host system [inherent to the system as a

bus interface is used to interface the controller with host computer];

a cache memory device [Fig. 1; RAM cache]; and

a non-volatile memory device, for storing logic code associated with the DSP, the first

interface and the second interface, wherein the logic code comprises instructions executable by

the DSP for maintaining a list of boot data used for booting the host system [fig. 1; col. 4, lines

10-28; instructions are embodied as microcode in a ROM; col. 5, lines 1-7; a prefetch table is

read from a reserved area of the disks], for preloading the boot data into the cache memory

device prior to completion of initialization of the central processing unit of the host system [col.
1, lines 58-64; col. 2, lines 36-41; col. 3, lines 30-39; col. 5, lines 17-21; data is preloaded into
the RAM cache according to the prefetch table prior to completion of initialization of the central
processor unit as shown in Fig. 3 that the method enters an idle state to await a command from
the host computer since the CPU of the host computer is not ready], and servicing requests for
boot data from the host system after completion of initialization of the central processing unit of
the host system using the preloaded boot data [col. 2, lines 41-47; col. 3, lines 30-39; data is
communicated from the cache to the host computer as soon as the host computer requests the
data upon completion of initialization of the CPU].

Kroeker et al do not disclose about accessing compressed boot data and decompressing
the compressed boot data. However, Esfahani et al clearly disclose about loading a compressed
boot data into a RAM cache and then the boot data is decompressed and executed [col. 2, lines 5-
13, 63, 67; col. 10, line 65 – col. 11, line 4]. Therefore, it would have been obvious to one of
ordinary skill in the art at the time the invention was made to combine the cited references as
both are directed to minimize a computer's initial program load time or shortening the load time
of the computer programs from a hard disk drive to a host computer. Moreover, the shortening
load time method of Kroeker et al by loading the program codes into the RAM cache according
to the prefetch table will definitely be benefited with the method of reading compressed data into
the RAM cache and then decompressing and executing as needed. This way, one may not only
have needed data into a fast access memory but also a large amount of data to avoid frequent
accessing the storage device(s).

12.   As per claim 15, Kroeker et al teach that the logic code in the non-volatile memory

device further comprises program instructions executable by the DSP for maintaining a list of

application data associated with an application program [col. 11; lines 30-34; a prefetch table

containing disk storage location and length of the data records requested by the application

program]; preloading the application data upon launching the application program [col. 11, lines

46-50; preloading the data cache prior to receiving a read command from the application], and

servicing requests for the application data from the host system using the preloaded application

data col. 11, lines 51-57; communicating the prestored data records of the application from the

data cache to the host computer].

13.   As per claim 17, Kroeker et al teach

        maintaining a list of application data associated with an application program [col. 2, lines

30-47; col. 5, lines 1-7; a prefetch table containing a listing of the disk locations and length of

data records that were requested by the host computer in the immediately previous power-

on/reset; col. 5, lines 41-51; requesting data records are part of a computer program such as DOS

or Windows; claims 28 and 32];

preloading the application data into the cache memory prior to completion of

initialization of the central processing unit of the computer system, wherein preloading the

application data comprises accessing application data from a boot device [col. 1, lines 58-64; col.

2, lines 36-41; col. 3, lines 30-39; col. 5, lines 17-21; data is preloaded into the RAM cache

according to the prefetch table prior to completion of initialization of the central processor unit

as shown in Fig. 3 that the method enters an idle state to await a command from the host

computer since the CPU of the host computer is not ready]; and

servicing requests for application data from the computer system using the preloaded

application data after completion of initialization of the central processing unit of the computer

system, wherein servicing requests comprises accessing application data from the cache [col. 2,

lines 41-47; col. 3, lines 30-39; data is communicated from the cache to the host computer as

soon as the host computer requests the data upon completion of initialization of the CPU].

Response to Arguments

14.   Applicant's arguments filed 05/02/2005 have been fully considered but they are not

persuasive.

15.   In the remarks, applicants argued in substance that (1) Kroeker does not disclose

preloading the boot data into a cache memory <u>prior to completion of</u> initialization of <u>a central</u>

<u>processing unit of</u> the computer system; (2) Kroeker does not teach a method for accelerated

loading of an application program.

16.   As to point (1), Kroeker clearly disclose preloading the boot data into a cache memory

prior to completion of initialization of a central processing unit of the computer system [col. 1,

lines 58-64; disk completes its booting process before the host computer is ready for program

transfer; col. 2, lines 36-41; col. 3, lines 30-39; col. 5, lines 17-21; data is preloaded into the

RAM cache according to the prefetch table prior to completion of initialization of the central

processor unit as shown in Fig. 3 that the method enters an idle state to await a command from

the host computer since the CPU of the host computer is not ready].

17.   As to point (2), Kroeker clearly disclose a method for accelerated loading of an

application program as claimed by Kroeker in claims 28 and 32. Kroeker expressly indicates

about requests data records of an application program [col. 11, lines 27-29; col. 12, lines 25-28].

Kroeker expressly discloses another object of the present invention for rapidly communicating a

computer program from a disk drive to a host computer [col. 2, lines 1-5].

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Suresh K. Suryawanshi whose telephone number is 571-272-3668. The examiner can normally be reached on 9:00am - 5:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Thomas C. Lee can be reached on 571-272-3667. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

sks
May 27, 2005

CHUN CAO